



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/527,861	03/16/2005	Christian Hentschel	NL 020911	2056
24737 7590 10/01/2008 PHILIPS INTELLECTUAL PROPERTY & STANDARDS P.O. BOX 3001 BRIARCLIFF MANOR, NY 10510				
EXAMINER				
TSAL SHENG JEN				
ART UNIT		PAPER NUMBER		
2186				
MAIL DATE		DELIVERY MODE		
10/01/2008		PAPER		

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary

Application No.

10/527,861

Applicant(s)

HENTSCHEL, CHRISTIAN

Examiner

SHENG-JEN TSAI

Art Unit

2186

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 14 August 2008.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1,3,4,6-9,11,12 and 14-17 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1,3,4,6-9,11,12 and 14-17 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 16 March 2005 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____
- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date _____
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: _____

DETAILED ACTION

1. This Office Action is taken in response to Applicants' Request for Continued Examination (RCE) filed on August 14, 2008 regarding application 10/527,861 filed on March 16, 2005.

2. Claims 1, 3-4, 6-9, 11-12 and 14-17 have been amended.

Claims 2, 5, 10 and 13 have been cancelled.

Claims 1, 3-4, 6-9, 11-12 and 14-17 are pending for consideration.

3. ***Response to Remarks and Amendments***

Applicants' remarks have been fully and carefully considered with examiner's response set forth below.

(1) Applicants contend that the Rodrigues reference fails to teach "a reallocation of the resources of the system such that the pop-up has the current focus of the user." The Examiner disagrees.

Rodrigues teaches [Advertisement pop-ups are optional supplements that enable the user to receive product information during the on-demand video presentation in exchange for a lower rental fee. The advertisements may be specific to products inherent during the course of the on-demand movie such as a watch worn by a leading actor. Furthermore, the advertisement supplement's active time interval may be concurrent to the time in the movie that the leading actor reads the time on the watch (column 10, lines 12-20)]. Thus, the system needs to, at the very least, allocate resources to ensure that the advertisement supplement's active time interval may be concurrent to the time in the movie that the leading actor reads the time on the watch.

Further, since the pop-ups are optional supplements to the user's current focus (for example, the movie would be the user's current focus when the user is watching a movie), resources have to be allocated so that this optional supplements, which may not exist in a basic setting, are added onto the basic stuff.

(2) Applicants amend claims 8 and 16 with additional new limitations.

In response, a new reference (Koski et al., US 5,596,502) has been identified and has been combined with the previously relied on references (Kraft and Rodriguez) to address all the limitations recited in claims 8 and 16.

Refer to the corresponding sections of the following claim analysis for details.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

5. Claims 1, 3-4, 6-7, 9, 11-12, 14-15 and 17 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kraft, IV et al. (US 6,091,414, hereinafter referred to as Kraft), in view of Rodriguez et al. (US 7,200,857, hereinafter referred to as Rodriguez).

It is noted that, in the following claim analysis, those elements recited by the claims are presented using **bold font**.

As to claim 1, Kraft discloses **a method** [System and Method for Cross-Environment Interaction in a Computerized Graphical Interface Environment (title)] **of allocating shared resources** [the shared resources includes the displaying screen

(figure 3 shows that two windows (30 and 35) associated two different applications are sharing the same display screen (26) of the computer system) and CPU time (this, in turn, will cause the system to increase CPU utilization of task 1 relative to task 2, column 7, lines 41-42)] **between applications with media information on a resource limited platform** [the X Window System environment permits execution of applications in the context of various windows (column 1, lines 63-64); A system and method for automatically adjusting priority assigned to execution of applications, tasks, or workspaces (abstract); figure 3 shows that two windows (30 and 35) associated two different applications are sharing the same display screen (26) of the computer system; When a given application such as X application 24 goes into and out of "focus", this may be visually represented by an associated window such as window 30 in display 26, by means of the border of the window changing color, and the window associated with the particular application coming to the top or being placed lower in the stacking order, respectively (column 5, lines 58-64)], **characterized in that the method comprises the following steps:**

identifying an application with a current focus of a user [In response to this user input, detected by the X Server 28, the X Server will communicate this input to the Window Manager 22, thereby indicating that window 35 has gained focus (column 7, lines 53-56)];

setting or increasing the allocation of resources for the application with the current focus of the user [it will be recalled that an additional feature of the invention is to provide for additional CPU 10 utilization for the task, application, or suite of

applications associated with that "focused" window (column 5, line 67 to column 6, line 3); The amount of CPU resource then directed to the particular application as a result of the priority alteration is thereby in turn altered. In this manner, a focused application is dynamically provided with more CPU resource relative to remaining tasks, applications, or suites thereof associated with a workspace executing in the multitasking environment (column 4, lines 17-23)]; and automatically allocating a remaining part of the resources to at least one application without the current focus of the user [Thus, the system in combination with the software steps represented in FIG. 4 will cause the appearance of a window associated with the task 1 to be changed and brought to the foreground, thereby signifying that the task 1 is focused. The system will then set the focus of task 1. This, in turn, will cause the system to increase CPU utilization of task 1 relative to that of task 2. Next, the system will alter the windowed appearance of the task 2 and move it to the background. Next, the system will clear focus of the previously selected task 2 and its corresponding window. CPU utilization for task 2 will then be decreased to that of task 1 prior to its selection (column 7, lines 36-47); hence, task 1 is in focus and task 2 is out of focus, but task2 is still allocated to certain amount of CPU time, and task 2 is also allocated to be displayed in the background of the screen], **wherein the step of identifying the application with the current focus of the user is selected from at least one of the group of: user controlled, system controlled, or externally controlled** [Kraft teaches that the current focus of the user is selected, at least, via user controller mechanism: It will first be assumed that a user input is typically provided by means of

the keyboard 34 or mouse 36, reflecting that a particular window and its associated task is of more interest or priority to the user. This is shown graphically in FIG. 3 by the arrow 33, indicating that a cursor of the mouse pointing device 36 for example, has been moved by a user into the window 35. This indicates that the end user desires to cause the window 35 to be "focused" into prominence both visually in the display 26 and in terms of the processing power allocated to tasks associated with this window 35 by the CPU 10 (column 6, lines 55-65); First it is assumed that the user has moved the pointing device, 51, whereby a window is either selected, 52, or deselected, 57. Assuming a window was selected, the window manager will then set focus (column 7, lines 12-15); In response to this user input, detected by the X Server 28, the X Server will communicate this input to the Window Manager 22, thereby indicating that window 35 has gained focus (column 7, lines 53-56)].

As to claim 1, Kraft does not teach that **a provider of the media information performs the externally controlled step of identifying the application with the current focus of the user.**

However, Rodriguez teaches in the invention "Synchronized Video-On-Demand Supplemental Commentary" a scheme for allowing multiple applications/windows to share display screen [The window manager 59 provides a mechanism for implementing the sharing of the screen regions and user input (column 5, lines 28-30)], which is very similar to the invention disclosed by Kraft.

Specifically, Rodriguez teaches **a provider of the media information performs the externally controlled step of identifying the application with the current focus**

of the user [Advertisement pop-ups are optional supplements that enable the user to receive product information during the on-demand video presentation in exchange for a lower rental fee. The advertisements may be specific to products inherent during the course of the on-demand movie such as a watch worn by a leading actor. Furthermore, the advertisement supplement's active time interval may be concurrent to the time in the movie that the leading actor reads the time on the watch (column 10, lines 12-20); note that Advertisement pop-ups are externally controlled and provided by a media provider].

Rodriguez also teaches that the motivation of allowing a provider of the media information performs the externally controlled step of identifying the application with the current focus of the user is to give users the option of saving [Advertisement pop-ups are optional supplements that enable the user to receive product information during the on-demand video presentation in exchange for a lower rental fee (column 10, lines 12-15)].

Therefore, it would have been obvious for one of ordinary skills in the art at the time of Applicants' invention to allow a provider of the media information performs the externally controlled step of identifying the application with the current focus of the user, as demonstrated by Rodriguez, and to incorporate it into the existing scheme disclosed by Kraft, in order to give users more options and more satisfaction.

As to claim 3, Kraft teaches **the method as claimed in claim 1, characterized in that the user controlled step of identifying the application with the current focus of the user, comprises one or more of the following steps:**

selecting a new application as the application with the current focus of the user, when the new application is opened [In an effort to address this deficiency regarding distributive applications, in yet an alternate implementation, newly built (e.g., non-legacy) applications may be linked to an Xt Intrinsic library capable of reprioritizing themselves as they leave or enter focus (column 9, lines 24-28)];

changing the application with the current focus of the user to an application just switched to upon switching to an already opened application [It will first be assumed that a user input is typically provided by means of the keyboard 34 or mouse 36, reflecting that a particular window and its associated task is of more interest or priority to the user. This is shown graphically in FIG. 3 by the arrow 33, indicating that a cursor of the mouse pointing device 36 for example, has been moved by a user into the window 35. This indicates that the end user desires to cause the window 35 to be "focused" into prominence both visually in the display 26 and in terms of the processing power allocated to tasks associated with this window 35 by the CPU 10 (column 6, lines 55-65)];

when the user closes down an application with the current focus, switching to the application with the preceding focus of the user by keeping a record of the order of previously opened applications to indicate their importance in decreasing order, where the most recently opened application has the highest importance [Thus, the system in combination with the software steps represented in FIG. 4 will cause the appearance of a window associated with the task 1 to be changed and brought to the foreground, thereby signifying that the task 1 is focused. The

system will then set the focus of task 1. This, in turn, will cause the system to increase CPU utilization of task 1 relative to that of task 2. Next, the system will alter the windowed appearance of the task 2 and move it to the background. Next, the system will clear focus of the previously selected task 2 and its corresponding window. CPU utilization for task 2 will then be decreased to that of task 1 prior to its selection (column 7, lines 36-47); When a given application such as X application 24 goes into and out of "focus", this may be visually represented by an associated window such as window 30 in display 26, by means of the border of the window changing color, and the window associated with the particular application coming to the top or being placed lower in the stacking order, respectively. In addition to this window manager 22 setting or clearing focus so as to cause the GUI window of the related application to come to the top of or being placed within the display stack 26 (column 5, lines 58-66)].

As to claim 4, Kraft teaches **the method as claimed in claim 1, characterized in that the system controlled step of identifying the application with the current focus of the user, is performed by one of the following steps:**

an automatically changing of the current focus of the user according to a predetermined priority hierarchy of the available applications [A system and method are provided for automatically adjusting priority assigned to execution of applications, tasks, or workspaces to thereby improve performance relative to other such applications, tasks or workspaces in a computerized multitasking graphical user interface environment (column 3, lines 61-66); The amount of CPU resource then directed to the particular application as a result of the priority alteration is thereby in

turn altered. In this manner, a focused application is dynamically provided with more CPU resource relative to remaining tasks, applications, or suites thereof associated with a workspace executing in the multitasking environment (column 4, lines 17-23)]; **keeping a record of the order of previously opened applications to indicate their importance in decreasing order, where the most recently opened application has the highest importance and, switching to the application with the preceding focus of the user when the user closes down an application with the current focus** [the record includes display stack (figure 3, 26) and window stacks; When a given application such as X application 24 goes into and out of "focus", this may be visually represented by an associated window such as window 30 in display 26, by means of the border of the window changing color, and the window associated with the particular application coming to the top or being placed lower in the stacking order, respectively. In addition to this window manager 22 setting or clearing focus so as to cause the GUI window of the related application to come to the top of or being placed within the display stack 26 (column 5, lines 58-66); Thus, the system in combination with the software steps represented in FIG. 4 will cause the appearance of a window associated with the task 1 to be changed and brought to the foreground, thereby signifying that the task 1 is focused. The system will then set the focus of task 1. This, in turn, will cause the system to increase CPU utilization of task 1 relative to that of task 2. Next, the system will alter the windowed appearance of the task 2 and move it to the background. Next, the system will clear focus of the previously selected task 2

and its corresponding window. CPU utilization for task 2 will then be decreased to that of task 1 prior to its selection (column 7, lines 36-47)].

As to claim 6, Kraft teaches **the method as claimed in claim 1, characterized in that the step of setting or increasing the output quality of the application with current focus of the user is performed automatically by means of automatic settings of the overall system control and with no additional input from the user** [A system and method are provided for automatically adjusting priority assigned to execution of applications, tasks, or workspaces to thereby improve performance relative to other such applications, tasks or workspaces in a computerized multitasking graphical user interface environment (column 3, lines 61-66); The amount of CPU resource then directed to the particular application as a result of the priority alteration is thereby in turn altered. In this manner, a focused application is dynamically provided with more CPU resource relative to remaining tasks, applications, or suites thereof associated with a workspace executing in the multitasking environment (column 4, lines 17-23); When a given application such as X application 24 goes into and out of "focus", this may be visually represented by an associated window such as window 30 in display 26, by means of the border of the window changing color, and the window associated with the particular application coming to the top or being placed lower in the stacking order, respectively (column 5, lines 58-64)].

As to claim 7, Kraft teaches **the method as claimed in claim 1** [refer to "As to claim 1"], **characterized in the step of decreasing or increasing the allocation of resources for the application with the current focus of the user is performed**

manually by user interaction by means of a user interface [It will first be assumed that a user input is typically provided by means of the keyboard 34 or mouse 36, reflecting that a particular window and its associated task is of more interest or priority to the user. This is shown graphically in FIG. 3 by the arrow 33, indicating that a cursor of the mouse pointing device 36 for example, has been moved by a user into the window 35. This indicates that the end user desires to cause the window 35 to be "focused" into prominence both visually in the display 26 and in terms of the processing power allocated to tasks associated with this window 35 by the CPU 10 (column 6, lines 55-65)].

As to claim 9, it recites substantially the same limitations as in claim 1, and is rejected for the same reasons set forth in the analysis of claim 1. Refer to "As to claim 1" presented earlier in this Office Action for details.

As to claim 10, it recites substantially the same limitations as in claim 2, and is rejected for the same reasons set forth in the analysis of claim 2. Refer to "As to claim 2" presented earlier in this Office Action for details.

As to claim 11, it recites substantially the same limitations as in claim 3, and is rejected for the same reasons set forth in the analysis of claim 3. Refer to "As to claim 3" presented earlier in this Office Action for details.

As to claim 12, it recites substantially the same limitations as in claim 4, and is rejected for the same reasons set forth in the analysis of claim 4. Refer to "As to claim 4" presented earlier in this Office Action for details.

As to claim 13, it recites substantially the same limitations as in claim 5, and is rejected for the same reasons set forth in the analysis of claim 5. Refer to "As to claim 5" presented earlier in this Office Action for details.

As to claim 14, it recites substantially the same limitations as in claim 6, and is rejected for the same reasons set forth in the analysis of claim 6. Refer to "As to claim 6" presented earlier in this Office Action for details.

As to claim 15, it recites substantially the same limitations as in claim 7, and is rejected for the same reasons set forth in the analysis of claim 7. Refer to "As to claim 7" presented earlier in this Office Action for details.

As to claim 17, Kraft teaches **a computer-readable medium having stored thereon instructions for causing a processing unit to execute the method as claimed in claim 1** [This invention relates to computer systems supporting graphical user interface environments and, more particularly, relates to systems and methods for obtaining information and effecting application control across dissimilar computer system boundaries (column 1, lines 14-18); FIG. 4 is a flow diagram implementable in software executing on the system of FIG. 3 illustrating the steps of the method of the invention (column 4, lines 37-39)].

6. Claims 8 and 16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kraft, IV et al. (US 6,091,414, hereinafter referred to as Kraft), in view of Rodriguez et al. (US 7,200,857, hereinafter referred to as Rodriguez), and further in view of Koski et al. (US 5,596,502, hereinafter referred to as Koski).

As to claim 8, Kraft in view of Rodriguez teaches **the method as claimed in claim 6** [refer to “As to claim 6”], but does not teach that **the automatic settings of the overall system control is influenced by a learning function, which takes previous user settings of the past into account, wherein the learning function is implemented as at least one of an averaging function, a recursive function, a non-recursive function, a non-linear function, a function with different weights, having the previous user settings as an input.**

However, Koski teaches in the invention “Computer System Including means for Decision Support Scheduling” a scheme for allocating the best available resources to produce products [in response to demand placed on the Cube World by a customer order, which program allocates the best available resources to produce products (abstract)], where a learning function is used [This invention relates generally to computerized management systems, and more particularly to computer-controlled closed loop multiple activity management coordinating systems that support effective resource utilization decision making and alternate path non-deterministic process development, including rule based learning functions to efficiently allocate resources to resolve conflicts in competition for the availability of limited resources to achieve multiple objectives scheduled in time (col. 1, lines 19-28); Learning-based : ... (col. 9, line 60 to col. 10, line 15)], and the learning function comprises a function with different weights [However, as the Cube System operates within the plant and receives prime data, it learns that skilled operators tend to assign certain types of products to certain machines and it adjusts the weighted rankings for those paths and learns to load

specific machines with those types of jobs (col. 9, line 65 to col. 10, line 10); It examines all possible alternate paths and ranks them based on desirability weights.

These desirability weights may be arbitrarily assigned by the user but usually will be learned by the system as history patterns, retained as Prime Data, are recognized by a neural network built into Cube Objects (col. 20, lines 3-9)].

Koski also teaches that the motivation of having a learning function with different weights to take previous user setting into account when performing automatic setting is to allocate the best available resources [in response to demand placed on the Cube World by a customer order, which program allocates the best available resources to produce products (abstract)].

Therefore, it would have been obvious for one of ordinary skills in the art at the time of Applicants' invention to have a learning function with different weights to take previous user setting into account when performing automatic setting, as demonstrated by Koski, and to incorporate it into the existing scheme disclosed by Kraft in view of Rodriguez, in order to provide customized display according to user's preference.

As to claim 16, it recites substantially the same limitations as in claim 8, and is rejected for the same reasons set forth in the analysis of claim 8. Refer to "As to claim 8" presented earlier in this Office Action for details.

7. Related Prior Art

The following list of prior art is considered to be pertinent to applicant's invention, but not relied upon for claim analysis conducted above.

- Wright, (US 6,204,847), "Shared Virtual Desktop Collaboratively Application System."
- Dubrow et al., (US 6,570,590), "Application Sharing in a Frame."
- Brenner et al., (US 6,584,488), "Controlling Allocation of System Resources with an Enhanced Priority Calculation."
- Rolia, (US Patent Application Publication 2003/0093527), "Method and System for Exploiting Service Level Objectives to Enable Resource Sharing in a Communication Network Having a Plurality of Application Environments ."

Conclusion

8. Claims 1, 3-4, 6-9, 11-12 and 14-17 are rejected as explained above.
9. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Sheng-Jen Tsai whose telephone number is 571-272-4244. The examiner can normally be reached on 8:30 - 5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Matthew Kim can be reached on 571-272-4182. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Sheng-Jen Tsai/

TFSA Examiner, Art Unit 2186

September 22, 2008